**Plan de Gestión**

**Proyecto EvaEduca: Plataforma educativa con asistente virtual y autoevaluación AI**

**Versión 1.2**

**Historial de Revisiones**

| **Fecha** | **Versión** | **Descripción** | **Autor** |
| --- | --- | --- | --- |
| 4/05/2024 | 1.0 | Versión preliminar como una propuesta de desarrollo. | Jefe de proyecto |
| 13/05/2024 | 1.1 | Desarrollo del documento | Jefe de proyecto |
| 23/05/2024 | 1.2 | Modificación de contenido: Definición de nomenclatura, clasificación | Jefe de proyecto |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Índice**

1. Introducción
2. Gestión de Documentos
   1. Definición de roles:
   2. Herramientas de gestión de la configuración:
   3. Herramienta elegida:
3. Actividades de la Gestión
   1. Identificación:
      1. Clasificación:
      2. Definición de nomenclatura:
      3. Diseño de repositorio:
      4. Línea base:
4. Referencias
5. Anexos
6. **Introducción**

IrisTech es una empresa de tecnología fundada en el año 2010, con más de una década de experiencia en el desarrollo de soluciones innovadoras para el sector educativo. Nos especializamos en proyectos que buscan mejorar la calidad y la eficiencia en la enseñanza y el aprendizaje, utilizando la tecnología como herramienta fundamental para lograrlo.

Durante nuestra trayectoria, hemos trabajado en una amplia variedad de proyectos, desde sistemas de gestión educativa hasta plataformas de aprendizaje en línea y aplicaciones móviles para la educación. Nuestro enfoque se centra en entender las necesidades específicas de nuestros clientes y diseñar soluciones a medida que se adapten a sus requerimientos y objetivos.

Sin embargo, a lo largo de nuestros proyectos, hemos enfrentado desafíos recurrentes que han afectado el éxito y la eficiencia de nuestras implementaciones. Problemas como la falta de comunicación efectiva, la gestión inadecuada de los recursos y la falta de alineación entre los equipos han sido obstáculos que hemos enfrentado en el pasado y que buscamos abordar de manera proactiva en el futuro.

Por esta razón, nos hemos propuesto implementar las mejores prácticas de Gestión de Proyectos para garantizar el éxito de nuestro próximo proyecto, EvaEduca. Nuestro objetivo es aplicar métodos y procesos estandarizados que nos permitan gestionar de manera eficiente los recursos, minimizar los riesgos y asegurar la entrega oportuna y exitosa de la plataforma educativa. Con este enfoque, esperamos superar los desafíos pasados y alcanzar nuevos niveles de excelencia en la industria de la tecnología educativa.

1. **Gestión de Documentos**
   1. **Definición de roles:**

| **Rol** | **Encargado** | **Descripción** |
| --- | --- | --- |
| **Gestor de la configuración** | Jefe de proyecto, arquitecto de software | Responsable principal de planificar, coordinar y supervisar todas las actividades relacionadas con la gestión de la configuración de software. Se asegura de que se sigan los procesos y procedimientos establecidos. |
| **Bibliotecario de configuración** | Jefe de proyecto | Define y da mantenimiento a las bibliotecas que son usadas durante  la gestión de configuración.  Es el encargado de asegurarse que los aspectos prácticos de la  gestión de configuración trabajen entre sí adecuadamente. |
| **Auditor de configuración** | Jefe de proyecto, analista funcional | Realiza revisiones periódicas para garantizar que se cumplan los estándares y procedimientos de gestión de la configuración. Identifica áreas de mejora y asegura la conformidad con los requisitos. |
| **Especialista en control de cambios** | Jefe de proyecto, analista funcional, testers | Responsable de evaluar y aprobar cambios propuestos en la configuración. Evalúa el impacto de los cambios, asegura la calidad de las modificaciones y aprueba su implementación. |

* 1. **Herramientas de gestión de la configuración:**

| **Herramienta** | **Fuente** | **Características** |
| --- | --- | --- |
| **GitHub** | [**https://github.com/**](https://github.com/) | - Control de versiones |
| - Gestión de ramas y fusiones |
| - Seguimiento de problemas (issues) |
| - Integración con otros servicios (CI/CD, Notificaciones) |
| - Colaboración y revisión de código |
| - Gestión de permisos y acceso |
| **GitLab** | [**https://about.gitlab.com/**](https://about.gitlab.com/) | - Control de versiones |
| - Gestión de ramas y fusiones |
| - Seguimiento de problemas (issues) |
| - Integración con otros servicios (CI/CD, Notificaciones) |
| - Colaboración y revisión de código |
| - Gestión de permisos y acceso |
| **Bitbucket** | [**https://bitbucket.org/**](https://bitbucket.org/) | - Control de versiones |
| - Gestión de ramas y fusiones |
| - Seguimiento de problemas (issues) |
| - Integración con otros servicios (CI/CD, Notificaciones) |
| - Colaboración y revisión de código |
| - Gestión de permisos y acceso |
| **SVN (Subversion)** | [**https://subversion.apache.org/**](https://subversion.apache.org/) | - Control de versiones |
| - Gestión de ramas y etiquetas |
| - Soporte para archivos binarios y grandes repositorios |
| - Seguridad y autenticación |
| - Gestión de permisos y acceso |
| **Mercurial (Hg)** | [**https://www.mercurial-scm.org/**](https://www.mercurial-scm.org/) | - Control de versiones |
| - Gestión de ramas y etiquetas |
| - Soporte para distribución de repositorios |
| - Rendimiento y eficiencia |
| - Gestión de permisos y acceso |
| **Azure DevOps** | [**https://azure.microsoft.com/es-es/**](https://azure.microsoft.com/es-es/) | - Control de versiones (Repositorios Git y TFVC) |
| - Integración con pipelines de CI/CD |
| - Seguimiento de problemas (boards) |
| - Integración con herramientas de desarrollo (VS Code, Eclipse) |
| - Gestión de permisos y acceso |

* 1. **Herramienta elegida: GitHub**

El siguiente diagrama explica de manera simplificada cómo utilizamos nuestra herramienta de control de versiones elegida para realizar operaciones con los ítems del repositorio de GitHub.

****

1. **Actividades de la Gestión**
   1. **Identificación:**
      1. **Clasificación:**

Como referencia hemos tomado nuestro cronograma de proyecto elaborado (Ver anexo 1), por lo tanto clasificaremos los ítems de acuerdo a su tipo de ítem, estos tipos de ítems pueden ser:

1. **Ítems de tipo evolutivo,** tales como documentos, los que están sujetos a una o más revisiones y nuevas liberaciones durante el ciclo de vida del software. Los Items en evolución son de dos tipos: Documentos, y archivos ejecutables o de soporte.
2. **Ítems de tipo fuente,** generalmente código fuente y archivos objeto utilizados para compilar una aplicación de software para ambiente de producción, los cuales son generalmente numerosos y cambian frecuentemente.
3. **Ítems de tipo soporte:** como sistemas operativos y software base, de los cuales el proyecto requiere ciertas versiones para su operación exitosa.

Para la clasificación de items, se tomó la siguiente regla de asignación:

* E = Ítem de tipo evolutivo
* F = Ítem de tipo fuente
* S = Ítem de tipo soporte

| **Tipo** | **Items** | **Fuente (E=Empresa, P=Proyecto, C=Cliente, V=Proveedor)** | **Extensión** | **Proyecto** |
| --- | --- | --- | --- | --- |
| E | Plan de Gestión de la Configuración de Software | E | .DOCX | - |
| E | Plan de Proyecto | P | .DOCX | EvaEduca |
| E | Cronograma del Proyecto | P | .XLSX |
| E | Documento de negocio | P | .DOCX |
| E | Documento de Especificación de Requisitos: Actores | P | .DOCX |
| E | Documento de Especificación de Requisitos: Casos de Uso | P | .DOCX |
| E | Documento de Especificación de UI | P | .DOCX |
| E | Documento de Guía de Estilos | P | .DOCX |
| E | Documento de Especificación de la BD | P | .DOCX |
| E | Documento de Arquitectura del Software | P | .DOCX |
| E | Reporte del Desarrollo del Software | P | .DOCX |
| F | Reporte del Primer Sprint | P | .DOCX |
| F | Reporte del Segundo Sprint | P |  |
| E | Manual de usuario | P | .DOCX |
| E | Documento de Pruebas del Software | p | .DOCX |
| F | Reporte del Tercer Sprint | P | .DOCX |
| F | Acta de cierre del proyecto | P | .DOCX |

* + 1. **Definición de nomenclatura:**

Hemos tomado los siguientes criterios o normas para gestionar adecuadamente nuestros ítems en el repositorio de GitHub con el propósito de mejorar el tiempo de respuesta de la identificación de ítems por parte del equipo:

* **Regla N°1** = "Acronimo del proyecto" + " - " + "Acronimo de Item"

Por ejemplo, **“**EE-DN” es la forma en que la regla de asignación nos permite expresar el ítem de documento de negocio (DN) que pertenece al proyecto EvaEduca(EE).

* **Regla N°2** : Cada línea base será denominada como LB (Acrónimo de Línea Base) seguido de su número correspondiente, por ejemplo:

*LB1 = Línea Base 1*

* **Regla N°3:** Las versiones y revisiones del documento se encontrarán dentro de cada uno en una tabla que especificará qué actualizaciones se han hecho hasta la fecha. Para identificar la versión y revisión utilizaremos identificadores numéricos separados por un punto; se presentaría de la siguiente forma:

**Versión.Revisión**

El número de versión cambiará cuando se modifique la arquitectura principal del ítem o cuando el ítem es completamente reconstruido. Asimismo, el número de revisión cambiará cuando el contenido haya cambiado pero no la estructura principal o el flujo del ítem.

* + 1. **Diseño de repositorio:**

La estructura que vamos a seguir para el diseño de repositorio será la siguiente:

* **Cliente:** Donde se encontrarán todos los ítems relacionados al cliente:
* **Desarrollo:** Donde se encontrarán todos los ítems para el desarrollo del proyecto:
  + Nombre de Proyecto
    - **Gestión:** Ítems necesarios para gestionar la organización del proyecto.
    - **Negocio:** Ítems relacionados a explicar el negocio o funcionamiento del proyecto.
    - **Análisis:** Ítems donde se analizarán los casos de uso, requisitos funcionales, no funcionales, etc.
    - **Diseño:** Ítems para evaluar el diseño de la base de datos, diseños UX/UI, entre otros relacionados.
* **Documentos**
* **Línea Base:** Donde se encontrarán todos los documentos aprobados de acuerdos los hitos del proyecto. El número de hitos define el número de líneas base que tendrá el proyecto.
  + Nombre de proyecto
    - **Línea base 1**
    - **Línea base 2**
    - **…**
    - **Línea base n**
    1. **Línea base:**

Para cada artefacto de cada hito en específico nos basamos en nuestro cronograma de proyecto (Ver anexo 1). Las líneas bases tendrán los siguientes artefactos:

| **Línea Base** | **Hito - Fecha** | **Artefactos** |
| --- | --- | --- |
| LB1 | 1 - 10/05/2024 | EE - PC, EE-CP, EE-RGH, EE-DN, EE-DERA, EE-DERCU-01, EE-DERCU-02, EE-DERCU-03, EE-DERCU-04, EE-DERCU-05, EE-DERCU-06, EE-DERCU-07, EE-DERCU-08, EE-DERCU-09, EE-DERCU-10, EE-DERCU-11, EE-DERCU-12, EE-DERCU-13, EE-DERCU-14, EE-DERCU-15, EE-DEUI, EE-DGE, EE-DEBD, EE-DAS, EE-RDS, EE-RPS. |
| LB2 | 2 - 20/06/2024 | Artefactos de la LB1, junto con nuevos documentos y actualización de documentos antiguos: EE-DER, EE-DEUI, EE-DEBD, EE-RDS, EE-RSS. |
| LB3 | 3 - 04/07/2024 | Artefactos de la LB2, junto con nuevos documentos y actualización de documentos antiguos: EE-DER, EE-DEUI, EE-DEBD, EE-DAS, EE-DGE, EE-MU, EE-DPS, EE-RDS, EE-RTS, EE-ACP. |

1. **Referencias:**

GitHub. (s.f.). Recuperado de https://github.com/

GitLab. (s.f.). Recuperado de https://about.gitlab.com/

Bitbucket. (s.f.). Recuperado de https://bitbucket.org/

Apache Subversion. (s.f.). Recuperado de https://subversion.apache.org/

Mercurial. (s.f.). Recuperado de https://www.mercurial-scm.org/

Microsoft Azure. (s.f.). Recuperado de https://azure.microsoft.com/es-es/

1. **Anexos:**
2. [EE - CP - Referencia](https://docs.google.com/spreadsheets/d/1O6We5cEsoDl1SKwUSzVfVcWAyT-x4hBt2O_8ngXO9AA/edit?usp=sharing)